

Индивидуальное задание #2. Строки в стиле C

Содержание отчета по лабораторной работе:

1. Заголовок (ФИО, группа, номер лабораторной, номер варианта)
2. Общая постановка задачи
3. Детальные требования и тест план
4. Результаты выполнения (исходные и результирующие файлы, скриншоты)
5. Код (файлы типа .cpp и .h)

Отчет загружается одним архивом.

Напишите функцию, использующую строки в стиле C и выполняющую действия в соответствии с указанным вариантом.

В функции **разрешается** использовать библиотечные функции из `<cctype>`, **но запрещается** пользоваться функциями из `<cstring>`.

Если функция формирует новую строку, то она должна иметь вид:

char * function(char *destination, const char *source, ...)

destination – строка, в которую помещается результат;

source – исходная строка (строки);

destination и source не должны перекрываться;

целевой массив (destination) должен содержать строку C и быть достаточно большим, чтобы вместить сформированную результирующую строку;

возвращаемое значение – указатель на строку результата (по аналогии с функциями из `<cstring>`)

Строка destination должна быть передана в функцию извне, чтобы **избежать** выделения динамической памяти внутри функции (по примеру стандартных функций из `<cstring>`).

При работе со строками в стиле C не следует передавать в функцию размер строки или вычислять его внутри функции специально. Все строки должны оканчиваться символом `'\0'` и нужно работать, учитывая это правило.

Для тестирования задания **напишите функцию main**, в которой строки будут считываться из файла, а результаты выводиться на экран и в файл.

Максимальную длину строки (строк) можно задать в коде константой, а сами символы строки (строк) вводить из файла. Данные для всех тестов могут быть в одном файле. Если в задаче требуется формирование строки-результата, то ее надо вывести в файл результата.

Для исключения повторных проходов по строкам можно использовать дополнительную память.

Результаты всех выполняемых тестов нужно вывести в один файл.

Варианты заданий:

- 1) Сформировать новую строку, удалив из исходной повторяющиеся пробелы (идущие подряд), а также пробелы в начале и в конце строки.
- 2) Сформировать новую строку, удалив из исходной строки все десятичные цифры.
- 3) Сформировать новую строку, удалив из исходной строки все буквы латинского алфавита.
- 4) Сформировать новую строку из символов, общих для двух исходных строк.
- 5) Сформировать новую строку из символов двух исходных строк, которые НЕ являются для них общими.
- 6) Сформировать новую строку из символов, которые в исходной строке повторяются более одного раза (в новой строке они должны встречаться по одному разу).
- 7) Определить, сколько различных букв латинского алфавита содержится в строке.
- 8) Определить, есть ли повторяющиеся символы в заданной строке. Функция должна возвращать true, если в строке есть повторяющиеся символы, false – в противном случае.
- 9) Определить, есть ли в заданной строке подряд стоящие одинаковые символы. Функция должна возвращать true, если такие символы в строке есть, false – в противном случае.
- 10) Сформировать новую строку, заменив в исходной строке все вхождения одного заданного символа на другой заданный символ.
- 11) Сформировать новую строку, заменив в исходной строке все прописные латинские буквы на строчные.
- 12) Сформировать новую строку, объединив две строки p и q по правилу: p[0], q[0], p[1], q[1], Исходные строки могут иметь разную длину, после завершения короткой строки дописать хвост более длинной.
- 13) Проверить, есть ли в двух заданных строках одинаковые символы. Функция должна возвращать true, если есть одинаковые символы, false – в противном случае.
- 14) Сформировать новую строку, содержащую все латинские буквы, отсутствующие в исходной строке. Заглавные и строчные буквы не различаются. Буквы новой строки должны следовать в порядке возрастания их кодов ASCII.
- 15) Сформировать новую строку из двух исходных, исключив из первой все символы, встречающиеся во второй.
- 16) Сформировать новую строку из двух исходных, добавив в первую все символы, встречающиеся во второй и являющиеся десятичными цифрами.
- 17) Определить, есть ли повторяющиеся цифры в заданной строке. Функция должна возвращать true, если в строке есть повторяющиеся цифры, false – в противном случае.
- 18) Сформировать новую строку, удалив из исходной строки все гласные буквы латинского алфавита.
- 19) Сформировать новую строку, состоящую из трёх чаще всего встречающихся символов исходной строки. Символы новой строки должны следовать в порядке возрастания их кодов ASCII.
- 20) Сформировать новую строку, содержащую все латинские буквы, присутствующие в двух заданных строках. Заглавные и строчные буквы не различаются. Буквы новой строки должны следовать в порядке возрастания их кодов ASCII.
- 21) Определить, сколько и какие гласные буквы латинского алфавита встречаются в строке.
- 22) Сформировать новую строку, выделяя часть исходной с заданной позиции до конца исходной строки.
- 23) Сформировать новую строку, выделяя заданное количество символов исходной строки, начиная с заданной позиции.